

DETAILED ACTION

1. Applicant's amendment dated October 19, 2009, responding to the Final Office action mailed July 17, 2009 provided in the rejection of claims 1, 2, 4-8, 16, and 17, wherein claim 4 has been amended.

EXAMINER'S AMENDMENT

2. An examiner's amendment to the record appears below. Should the changes and/or additions be unacceptable to applicant, an amendment may be filed as provided by 37 CFR 1.312. To ensure consideration of such an amendment, it **MUST** be submitted no later than the payment of the issue fee.

3. Authorization for this examiner's amendment was given in a telephone interview with Mr. Greeley (Reg. No. 31,019) on December 23, 2009 to further amend claims 1, 2, 4, 5, 16, and 17 (see Examiner's Amendment below) and thus to obviate any potential 35 U.S.C 112, second paragraph issues and to place the claims in the condition for allowance.

4. The application has been amended as follows:

IN THE CLAIMS,

Please amend claims 1, 2, 4, 5, 16, and 17 as follows:

1. (Currently amended) A source control system for a process control system, comprising:

a processor in a process control system;

a database accessible by said processor to store information associated with an object under source control to be checked-out; and

a check-out function operable on said processor (a) to check-out said object, wherein said object being checked out is a user defined template (UDT) that is derived from a preconfigured block type object and that inherits pre-defined parameters from said pre-defined block object, wherein any existing dependent objects are children (UDTs) of said object being checked out or instances of said object being checked out, wherein said instances are objects that inherit block type objects and parameters of said UDT or of said children UDTs, and wherein said stored information includes a reference to an existing parent object, (b) to propagate changes made to said object to any of said existing dependent objects, when said object is saved, (c) to use said information to determine whether any of said dependent objects exist and whether at least one parent object exists, and (d) to automatically check-out said existing dependent objects, wherein said stored information includes a reference to said existing parent object, wherein said object being checked out is a user defined

~~template that is derived from a preconfigured object, and wherein said existing dependent objects are children user defined templates of said object being checked out or instances of said object being checked out or of said children user defined templates.~~

2. (Currently amended) The system according to claim 1, wherein said changes are further propagated comprising:

~~a propagation function operable on said processor to propagate changes made to said object being checked out to said existing dependent objects and to said existing parent object, when said object being checked out is saved.~~

3. (Canceled)

4. (Currently amended) The system according to claim 1, wherein said stored information is at least one selected from the group consisting of: a name, a version number, a type and a status, and wherein said stored information is further used by said processor to prevent unauthorized changes to dependent objects under control of said source control system.

5. (Currently amended) A method of automatic check-out for a source control system in a process control system, comprising:

Art Unit: 2192

storing information associated with an object under source control to be checked-out;

receiving a request from a user to check-out said object;

determining whether any dependent objects of said object being checked out exist and whether any parent objects exist based on said information, wherein said object being checked out is a user defined template (UDT) that is derived from a preconfigured block type object and that inherits pre-defined parameters from said pre-defined block object, wherein any existing dependent objects are children (UDTs) of said object being checked out or instances of said object being checked out, wherein said instances are objects that inherit block type objects and parameters of said UDT or of said children UDTs, and wherein said stored information includes a reference to said existing parent objects;

propagating changes made to said object to any of said existing dependant objects, when said object is saved;

automatically checking-out said existing dependent objects when said object is checked-out, ~~wherein said object being checked out is a user defined template that is derived from a preconfigured object, and wherein said existing dependent objects are children user defined templates of said object being checked out or instances of said object being checked out or of said children user defined templates;~~ and providing a status to said user.

Art Unit: 2192

6. (Previously presented) The method according to claim 5, further comprising:

sorting said existing dependent objects so that said existing parents precede children.

7. (Previously presented) The method according to claim 5, wherein one of said existing dependent objects is a derivation child of said object being checked out.

8. (Original) The method according to claim 7, further comprising:
automatically checking-out said derivation child only if said derivation child is checked-in.

9-15. (Canceled)

16. (Currently amended) A computer readable medium having executable instructions stored thereon to perform a method of version control for a process control system, said method comprising:

when checking-in an object, wherein said checked-in object is a user defined template (UDT) that is derived from a preconfigured block type object and that inherits pre-defined parameters from said pre-defined block object, determining relationships of said object by:

determining whether said object being checked-in has a first derivation parent;

adding a name and a version of said first derivation parent to a list of object relationships, if said object being checked-in has said first derivation parent, wherein said first derivation parent is a first parent UDT of said object being checked-in of a first instance of said object being checked-in, wherein said first instance is an object that inherits block type objects and parameters of said UDT, or of said first parent UDT;

if said object being checked-in does not have said first derivation parent, determining ~~for~~ if each contained object, ~~which that~~ is contained in said object being checked-in, ~~whether said contained object~~ has a second derivation parent, ~~if said object being checked-in does not have said first derivation parent;~~

adding a name and a version of said second derivation parent to said list of object relationships, if said contained object being checked-in has said second derivation parent; and
providing said list of object relationships.

17. (Currently amended) A computer readable medium having executable instructions stored thereon to perform a method of automatic check-out for a source control system in a process control system, said method comprising

storing information associated with an object under source control to be checked-out;

receiving a request from a user to check-out said object;

determining whether any dependent objects of said object being checked-out exist and whether at least one parent object of said object exists based on said information, wherein said object being checked out is a user defined template (UDT) that is derived from a preconfigured block type object and that inherits pre-defined parameters from said pre-defined block object, wherein any existing dependent objects are children (UDTs) of said object being checked out or instances of said object being checked out, wherein said instances are objects that inherit block type objects and parameters of said UDT or of said children UDTs, and wherein said stored information includes a reference to said existing parent object;

propagating changes made to said object to any of said existing dependant objects, when said object is saved;

automatically checking-out said existing dependent objects when said object being checked-out is checked-out, ~~wherein said object being checked-out is a user defined template, and wherein said existing dependent objects are children user defined templates of said object being checked-out or instances of said object being checked-out or of said children user defined templates;~~ and

providing a status to said user.

- **END OF AMENDMENT** -

Allowable Subject Matter

5. Claims 1, 2, 4-8, 16, and 17 (renumbered as 1-9) are allowed.

6. The following is an examiner's statement of reasons for allowance:

The cited prior art taken alone or in combination fails to suggest

"A source control system for a process control system, comprising:

...

a check-out function operable on said processor (a) to check-out said object, wherein said object being checked out is a user defined template (UDT) that is derived from a preconfigured block type object and that inherits pre-defined parameters from said pre-defined block object, wherein any existing dependent objects are children (UDTs) of said object being checked out or instances of said object being checked out, wherein said instances are objects that inherit block type objects and parameters of said UDT or of said children UDTs, and wherein said stored information includes a reference to an existing parent object, (b) to propagate changes made to said object to any of said existing dependent objects, when said object is saved, (c) to use said information to determine whether any of said dependent objects exist and whether at least one parent object exists, and (d) to automatically check-out said existing dependent objects.", as recited in independent claims 1 and similarly recited in independent claims 16 and 17; and

"...

when checking-in an object, wherein said checked-in object is a user defined template (UDT) that is derived from a preconfigured block type object and that inherits pre-defined parameters from said pre-defined block object, determining relationships of said object by:

determining ... has a first derivation parent;

adding a name and a version of said first derivation parent to a list of object relationships, if said object being checked-in has said first derivation parent, wherein said first derivation parent is a first parent UDT of said object being checked-in of a first instance of said object being checked-in, wherein said first instance is an object that inherits block type objects and parameters of said UDT, or of said first parent UDT;

if said object being checked-in does not have said first derivation parent, determining ... has a second derivation parent;

adding a name and a version of said second derivation parent to said list of object relationships, if said contained object being checked-in has said second derivation parent; and
providing said list of object relationships.”, as recited in independent claims 16.

7. Claims (2, 4), and (6-8) are considered allowable by virtue of their dependence on allowable independent claims 1, and 5 respectively.

Art Unit: 2192

8. Any comments considered necessary by applicant must be submitted no later than the payment of the issue fee and, to avoid processing delays, should preferably accompany the issue fee. Such submissions should be clearly labeled "Comments on Statement of Reasons for Allowance."

Conclusion

9. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Ben C. Wang whose telephone number is (571) 270-1240. The examiner can normally be reached on 8:00-5:30 (EST/EDT), Monday through Friday.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q. Dam can be reached on (571) 272-3695. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300. Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197

Art Unit: 2192

(toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Ben C Wang/

Examiner, Art Unit 2192

/Michael J. Yigdal/

Primary Examiner, Art Unit 2192